



(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
05.12.2001 Bulletin 2001/49

(51) Int Cl.7: **H04L 1/18**

(21) Application number: **00111296.0**

(22) Date of filing: **25.05.2000**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE
 Designated Extension States:
AL LT LV MK RO SI

(72) Inventors:
 • **Sachs, Joachim**
52066 Aachen (DE)
 • **Meyer, Michael**
52062 Aachen (DE)

(71) Applicant: **TELEFONAKTIEBOLAGET LM**
ERICSSON (publ)
126 25 Stockholm (SE)

(74) Representative: **Tonscheidt, Andreas**
Ericsson Eurolab Deutschland GmbH
Research Department
Ericsson Allee 1
52134 Herzogenrath (DE)

(54) **Selective repeat protocol with dynamic timers**

(57) In a method for a transmission of data packets from a transmitter to a receiver, the receiver performs a first check whether a data packet is correctly received and sends a request for a retransmission of erroneous data packets to the transmitter. The receiver performs a second check whether a received data packet (DP) is

a retransmission. The result of the second check is evaluated to adapt an operating parameter of the receiver or determine an operating parameter and especially to determine the sending of further requests (STATUS') for a retransmission. A receiver and a program unit embodying the invention are also described.

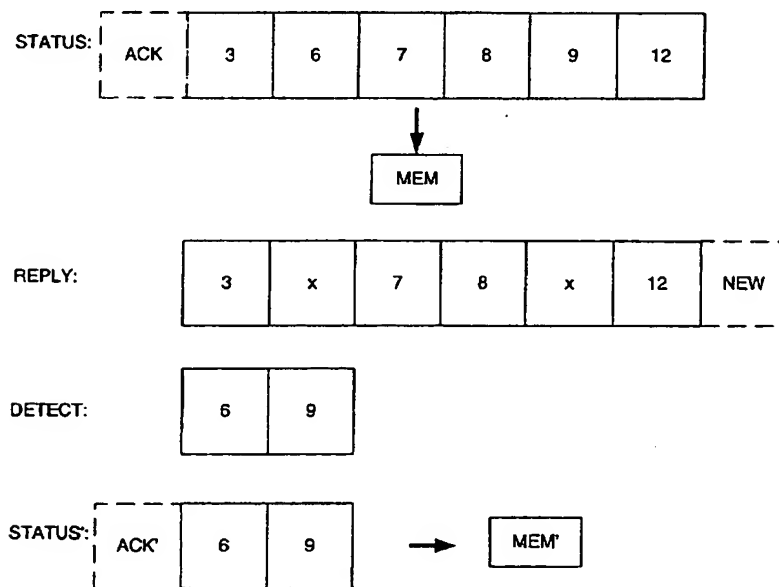


Fig. 3

Description

Technical field of the invention

[0001] The present invention relates to a method according to the preamble of claim 1. Devices and software programs embodying the invention are also described.

Background

[0002] When data packets are sent from a transmitter to a receiver, usually some packets are erroneous, i.e. not correctly received, due to transmission errors or loss. To cope with transmission errors and losses, ARQ (Automatic Repeat Request) protocols are used in most communication systems. In an ARQ protocol, the receiver checks whether a data packet is correctly received and sends a request for a retransmission of erroneous data packets to the transmitter. An example for an ARQ protocol which can be configured in a flexible way is the UMTS (Universal Mobile Telecommunication System) RLC (Radio Link Control) protocol as described in 3G TS 25.322 V3.2.0 of the 3rd Generation Partnership Project.

[0003] One optimization issue for ARQ protocols is to transmit Service Data Units (SDU) from a higher layer of the protocol stack used in the communication system in the shortest possible time between transmitter and receiver. For transmission, an SDU is in general segmented into several data packets of the transmission protocol, e.g. PDUs (Packet Data Units) according to the RLC protocol. Every erroneous data packet is retransmitted according to the ARQ configuration. An SDU can only be delivered to the higher layer when all data packets containing data of said SDU have arrived correctly. Therefore the SDU delay is determined by the data packet which needs the longest time for successful transmission, generally one of those data packets with the highest number of retransmissions. As the SDUs are generally forward in a specified sequence, the result is often a bursty transmission pattern on the layer which is stacked upon the ARQ protocol layer.

[0004] Many ARQ protocols have a mechanism which enables the receiver to detect if data packets are lost. For this purpose, data packets are identified by a sequence number which is assigned by a counter in the transmitter and incremented for each transmitted data packet. A state variable in the receiver keeps track of the highest received sequence number of a data packet (denoted VR(H) for RLC PDUs in 3G TS 25.322). Because the sequence number is generally limited to a maximum value, e.g. by the size of a corresponding data field in the data packet, a modulo operation is performed, i.e. the counter is reset when the maximum value is reached. Therefore the term "highest sequence number" refers throughout this description to the highest value of the sequence number before the modulo oper-

ation.

[0005] To avoid confusions due to the limited amount of sequence numbers, a reception window is defined with a lower limit denoted VR(R) and an upper limit denoted VR (MR) in 3G TS 25.322. Only data packets with sequence numbers in this interval are accepted by the receiver while all other data packets are discarded. The reception window is shifted according to the data packets received.

[0006] If data packets are transmitted in sequence and no reordering occurs on the transmission link, a transmission error can be detected by checking whether a sequence number is missing, i.e. whether a received sequence number is not equal to the last received sequence number plus the counter increment. As data packets identified as erroneous are preferably discarded without check of the sequence number, this implies that all data packets with a sequence number between the last and the present received sequence number have had an error or are lost. A similar mechanism is applied also in other ARQ protocols, e.g. SSCOP (Service Specific Connection Oriented Protocol). With this mechanism transmission errors can only be detected for the first transmission of a data packet while it is not possible to detect if any retransmission is erroneous. However, in particular retransmissions determine the SDU delay.

[0007] To request a retransmission, the receiver sends a status message to the sender, the message containing positive and negative acknowledgements of data packets. The transmitter then retransmits the requested data packets, i.e. those with a negative acknowledgement. These steps are repeated until all data packets have been received correctly. Both between two status messages and between the reception of a data packet and the subsequent status message, certain delay times exist. It is generally possible (e.g. by setting the STATUS prohibit timer in UMTS RLC) to specify these times, especially to avoid unnecessary retransmissions. Different triggers for a status message are possible. One common trigger for the transmission of a further status message is the detection of errors in data packets transmitted for the first time.

[0008] Alternatively or in addition, the Estimated PDU Counter (EPC) as described in 3G 25.322 can be used to estimate the arrival of retransmitted data packets from rate information, Transport Format Combination Identity (TFCI) and an estimated RLC round trip time. In the EPC method, the estimated round trip time of a message is stored in an EPC_timer, i.e. when the timer is expired, the first data packet retransmitted upon a request should arrive. With the reception of every received data packet, an EPC_counter is deducted which contains the number of requested data packets. Therefore, all requested packets should have arrived, when the EPC_counter is equal to zero. The EPC can then be used to trigger a further status message. However, the EPC method is sensitive to the precise determination of the round trip

time which can vary due to scheduling especially when MAC (Medium Access Control) multiplexing is used to combine several RLC channels on a common channel. On common channels, the round trip time is almost impossible to estimate. Furthermore, the user equipment identity and logical channel identity is not accessible on the RLC layer for common channels from the TFCI. So the EPC cannot be used on common channels and the uncertainty of the RLC round trip time degrades the EPC performance also on dedicated channels.

Summary and description of the invention

[0009] It is an object of the present invention to obviate the above disadvantages and improve the determination and adaptation of operating parameters in a transmission according to an ARQ protocol. It is a further object, to reduce the transmission time of data packets. It is still a further object, to reduce the burstiness of packets forwarded to higher layers in the protocol stack.

[0010] According to the invention, the method described in claim 1 is performed. Furthermore, the invention is embodied in a receiver as described in claim 15 and a software program as described in claim 16. Advantageous embodiments are described in the dependent claims.

[0011] In the proposed method, a transmission of data packets from a transmitter to a receiver is performed. The receiver performs a first check whether a data packet is correctly received and sends a request for a retransmission of erroneous data packets to the transmitter. The receiver performs also a second check whether a received data packet is a retransmission. The result of the second check is evaluated to adapt an operating parameter of the receiver or determine an operating parameter. For example the detection of retransmissions can be used to determine the round-trip time from the difference between the sending of the request and the reception of the retransmission. The performance of the protocol can be improved using the measured values to adapt operating parameters, e.g. timers, accordingly in the transmitter or receiver.

[0012] By precise adaptation of operating parameters, the transmission time of data packets can be reduced. As a result, also the SDU delay for protocol layers supplied with data packets by an ARQ protocol can be reduced. With increasing number of retransmissions, i.e. a high block error rate which is significantly large in wireless systems, the SDU delay also increases and the more significant is the improvement of the proposed method. Especially for a protocol with a delivering of SDUs in sequence, the burstiness of SDU delivery is reduced.

[0013] In contrast to EPC, the proposed method can also be used on common channels. It can be used independently of EPC. If it is applied combined with EPC, the uncertainty of the RLC round trip time can be reduced by stopping the EPC timer when the first retrans-

mission is detected and start decreasing the EPC counter at that instant.

[0014] In a preferable embodiment, a further request for a retransmission is initiated according to the result of the second check. The conditions for sending a further request can be determined by a parameter which is in turn adapted due to the result of the second check. For example, the retransmission can be initiated by a timer expiry or a trigger which is adjusted according to the second check.

[0015] In many protocols, data packets are identified by a sequence number. The length of the sequence number is preferably limited to improve the performance of the protocol. To avoid a confusion of data packets when the number of retransmissions is high, a reception window is defined by a first variable identifying the first data packet in the sequence requested for retransmission. A data packet is preferably rejected if the sequence number is smaller than the first variable or larger than the first variable plus an offset which determines a window size.

[0016] If data packets are identified by a sequence number, the receiver can store a further variable identifying the last data packet in the sequence which was correctly received. In this case, the second check is preferably performed by determining whether the sequence number of a received packet is smaller than the sequence number of said further variable. In this way, the processing is accelerated and a further processing of data packets which are detected as erroneous by the receiver and are to be discarded can be avoided.

[0017] In an advantageous embodiment, a timer or counter determining the sending of a further request for a retransmission is set according to the result of the second check. Timers or counters which can be adapted in this way to improve the precision of the timing are for example the timers Timer_EPC, Timer_Status_Prohibit or Timer_Status_Periodic as defined in 3G TS 25.322.

[0018] Preferably, a further check is performed whether a retransmitted data packet is erroneous. For this purpose, the results of the first and second check can be combined. If erroneous data packets are discarded after the first check, the further check preferably detects whether all packets requested for retransmission are successfully received.

[0019] It is proposed that said further check comprises a detection whether the sequence number of a successfully received packet is equal to the first variable determining the position of the reception window. In this case, the position of the reception window can be adapted.

[0020] It is proposed that at least one condition for sending a request for the retransmission of an erroneous retransmitted data packet differs from the conditions for sending a request for the retransmission of an erroneous data packet which was transmitted for the first time, i.e. which was not retransmitted yet. In this way, the transmission of the data packets most critical for de-

lay of SDUs can be accelerated while an unnecessary repeated transmission of data packets can be avoided.

[0021] In a preferable embodiment of the method, the sending of a request for retransmission, i.e. a status message, is triggered by the detection of an erroneous retransmitted data packet. It is possible to use a timer or counter to prohibit the sending until all data packets which were retransmitted are examined whether the retransmissions were erroneous.

[0022] A request for a further retransmission can be suspended while a retransmitted data packet is received. Preferably the request is suspended until all data packets requested for retransmission are received.

[0023] In an advantageous embodiment of the invention, the receiver stores which data packets are requested for retransmission, preferably by storing sequence numbers. As a result, retransmitted data packets can be identified by a simple comparison with the memory and especially the second check can be performed by comparison of stored and received sequence numbers.

[0024] To simplify the second check, the sequence number of the next requested data packet is determined, e.g. from the memory, and a retransmission is defined as erroneous if the sequence number of the next received data packet differs from the sequence number of the next requested data packet. By comparison of received and stored sequence numbers, a suspension of a further request for retransmissions can also easily be implemented.

[0025] The method is especially adapted for communication systems with a high error rate and large or variable round trip times. A typical example is a mobile communication system with data packet error rates on the order of several percent and round trip times between several 10 ms up to seconds.

[0026] A receiver in a mobile telecommunication system has a reception unit for receiving data packets from a transmitter. The receiver includes a processor for the detection of erroneous data packets and for initiating a request to the transmitter for a data packet detected as erroneous. The receiver can perform any of the above methods. The receiver can for example be part of a radio base station or user equipment in a mobile communication system.

[0027] It is preferable to embody the proposed method as a program unit which can be stored on a data carrier or is loadable into the processing system of a receiver. The unit comprises code for performing the steps of any of the above described methods when executed in a processing system.

[0028] The foregoing and other objects, features and advantages of the present invention will become more apparent in the following detailed description of preferred embodiments as illustrated in the accompanying drawings.

Brief description of the drawings

[0029]

- 5 Fig. 1 shows simulated results of a data transmission
- Fig. 2 depicts the reception window in a receiver
- Fig. 3 illustrates the handling of data packets according to the invention
- 10 Fig. 4 shows a preferable process for the handling of data packets according to the invention

Detailed description of the drawings

15 [0030] Figure 1 illustrates the problem of a bursty traffic pattern caused by the delay SDUs due to missing data packets on the RLC layer. Most ARQ (Automatic Repeat Request) protocols have an in-order SDU delivery. Therefore, PDUs which need to be retransmitted more than once delay not only the SDU for which they comprise data but also all subsequent SDU which may already be completely stored in the receiver. Because of the necessary storing and processing capacities for bursty traffic, it is disadvantageous in general and in particular for TCP (Transmission Control Protocol) traffic.

20 [0031] A simulation of a TCP receiver trace for a 384 kbit/s link in the state of the art is depicted in figure 1. As can be seen, RLC interactions do not lead to packet losses or TCP segment reordering for a reliable configuration of the link layer and in-order delivery of SDUs as used in the simulation. However, a bursty traffic pattern for received TCP segments is observable which is due to the ARQ mechanisms of the RLC layer. Missing RLC PDUs are retransmitted after the reception of a corresponding status message. The bundle of retransmitted PDUs and the in-order delivery causes the release of several TCP segments by the RLC at the same time, especially for high data rates. This effect is evident from the arrival of TCP segments in intervals of 100 ms which correspond to a STATUS prohibit timer set to 100ms, i. e. status messages are separated at least by this interval. Thus the periodicity of the status messages influences also the TCP layer. The bursty release of TCP segments is disadvantageous because it can cause problems with buffer overflows in the receiver or other entities, e.g. routers, to which the data is transferred from the receiver.

25 [0032] In figure 2, a reception window of a receiver for a protocol with packets arriving in a specified sequence is shown. The position of the reception window is determined by two state variables. The lower edge of the window corresponds to a first variable VRR containing the sequence number of the first data packet which has not been received yet. The variable VRR is updated when the corresponding data packet is received. The upper edge of the window is determined by the state variable VRM which is the value of variable VRR plus an offset, the offset being the window size. The receiver accepts

only data packets with a sequence number larger of equal to variable VRR and smaller than variable VRM while other data packets are discarded.

[0033] Within the window, a variable VRH indicates the highest expected sequence number of a data packet. A data packet with a sequence number $SN = VRH - 1$ is successfully received while data packets with the sequence number $SN = VRH$ or higher are not received yet. If the next data packet received has a sequence number larger than variable VRH, data packets with sequence numbers between VRH and the present number are erroneous. Data packets with a sequence number smaller than VRH were already transmitted to the receiver while data packets with a sequence number equal or larger than VRH are received for the first time. Variable VRH is updated with the reception of any data packet with a sequence number equal or larger.

[0034] In a typical situation, data packets are received and stored in the memory for the RLC layer of the receiver as indicated by the hatched data packets PD while other data packets have not been received yet. All packets with a sequence number smaller than VRR are successfully received and can be transferred for the assembly of SDUs for the next layer of the protocol stack. Between variables VRR and VRH, some data packets PD have been received while others are still missing. Between variables VRH and VRM, no data packets have arrived yet. For the proposed method, it is suitable to define a variable VRN denoting the next data packet expected for retransmission. In many situations, the values of variables VRR and VRN can be identical, e.g. during the transmission of data packets for the first time. However, when the last received retransmitted data packet is larger than VRR, e.g. data packet DP_R in the example, VRR and VRN can differ.

[0035] Basic ideas of the invention are to detect retransmissions, to find out which retransmissions are erroneous and to request retransmissions with low delay. It is possible to use the first of these steps only or the first and second step only to improve existing ARQ protocols. However, better protocol performance can be attained by a combination of all steps.

[0036] A detection of retransmissions can be used to improve the performance of timers which are used in the state of the art. It is possible to use several timers simultaneously in an ARQ protocol to configure the ARQ functions. Examples are the timers Timer_EPC, Timer_Status_Prohibit or Timer_Status_Periodic as defined in 3G TS 25.322. Preferably, the timers cause a separation of subsequent status messages by an interval which is equal to or slightly larger than the round trip time of the ARQ protocol. This avoids unnecessary retransmissions and allows at the same time a fast ARQ mechanism. By detecting retransmissions, it is possible to determine the round trip time by measuring the time a status message was sent and the time the first retransmission is received. The result can for example be an alternative to the Timer_EPC by beginning to decrease

the EPC counter when the first retransmission is detected.

[0037] The detection of erroneous retransmissions can be used to trigger the transmission of status messages from the receiver to the transmitter when missing retransmissions are detected while in the state of the art only missing new transmissions are detected. The proposed method can trigger status messages more quickly for the most delay critical PDUs, i.e. those which have been erroneous again in the retransmission.

[0038] Requesting the retransmission of erroneous data packets with low delay further reduces the ARQ SDU delay. The ARQ principle described above is differentiated for new transmissions and retransmissions. The delay according to the state of the art is preferably only used between status messages corresponding to the first transmissions of data packets whereas the delay is selected significantly smaller before a status message requesting the retransmission of data packets which were already erroneously retransmitted.

[0039] The execution of an advantageous embodiment of the proposed method is described with respect to figure 3. In ARQ protocols, retransmissions are preferably performed in order of increasing sequence numbers and retransmissions are transmitted before new data packets are sent, i.e. retransmissions have priority over new transmissions. This improves the performance of the protocol because the SDU delay is minimized and reception window stall conditions are avoided. Therefore, it is for example implemented in GPRS- (General Packet Radio Service), UMTS- and most other ARQ protocols. Priority of retransmissions over new transmissions is advantageous for the proposed embodiment although it may be preferable in some situations to perform some new transmissions before retransmissions, e.g. to clear a buffer.

[0040] When a message STATUS is sent, the receiver stores which data packets have been requested for retransmission in a memory MEM. In addition to the requested data packets, in the example the packets with the sequence numbers 3, 6, 7, 8, 9, 12, the message STATUS can comprise acknowledgements ACK for further data packets which were correctly received. After the message STATUS is sent, the receiver waits for the requested data packets. When requested data packets arrive in one or several response messages REPLY to the message STATUS, the receiver can detect retransmissions by comparison of the received and stored sequence numbers.

[0041] In the response messages REPLY, one or several data packets can be corrupted and are discarded as indicated by an x. Corrupted data can for example be detected by comparison of a check sum included in the data packet and a corresponding check sum calculated by the receiver. As the transmitter sends data packets in the same sequence as requested in the message STATUS, the receiver can immediately detect if retransmissions are lost. In the example, the receiver de-

tests from comparison of the stored message STATUS and the contents of the message reply that the data packets with sequence numbers 6 and 9 are missing as soon as the packets with sequence numbers 7 and 12 are received, respectively. Missing retransmissions are indicated separately in step DETECT. The messages REPLY may also include further data packets NEW which are transmitted for the first time.

[0042] A further message STATUS' for requesting the erroneous retransmissions indicated in step DETECT is sent. It is possible to use the same message STATUS as for ordinary requests for retransmission. Alternatively, a special message is used, e.g. comprising a bit indicating to the transmitter that the request concerns an erroneous retransmission which can for example be used to adapt parameters of the transmitter for handling the response message. If a mechanism is used to delay the sending of messages STATUS by the receiver (e.g. a STATUS prohibit timer), preferably different conditions are applied to the sending of a message STATUS'. It can for example be sent immediately after detection of an erroneous retransmission, after all retransmissions have arrived or according to an attributed timer. For a check of the reply to message STATUS' for erroneous retransmissions, the requested data packets are stored in a memory MEM'. Preferably, the contents of all messages STATUS, STATUS' are stored separately to simplify the handling.

[0043] Optionally, status information of data packets which were transmitted for the first time since the message STATUS can also be included into the message STATUS' as indicated by the acknowledgements ACK'. This can further accelerate the RLC performance if it is assured that this option does not interfere with the RLC configuration.

[0044] It is also possible, that erroneous retransmissions are requested with a regular message STATUS and not with a special message STATUS'. If a message STATUS is to be sent in this case - e.g. when a STATUS prohibit timer expires and a message STATUS has been triggered - the message STATUS is preferably delayed while retransmissions are received. The message STATUS is then sent when all retransmissions are received, so that it can include information about all erroneous retransmissions.

[0045] Many ARQ protocols are configured that the receiver does not send more than one STATUS message within a protocol round-trip time, e.g. to avoid unnecessary retransmissions. Because this assumption is not necessarily valid for the proposed method, retransmissions can not unambiguously be related to a specific message STATUS, STATUS'. The messages can be overlapping, i.e. have common areas of the receiver window on which they report. In this case, the information of all messages STATUS, STATUS' are stored in the receiver. When retransmissions arrive the memory contents for all messages are updated. By correlating received retransmissions with the open messages STA-

TUS, STATUS', it is in many cases possible to identify which message triggered the retransmissions, e.g. from parts of the STATUS messages which do not overlap with another message.

[0046] Figure 4 shows an example which is based on state variables as used in 3G TS 25.322. In addition, a new state variable VRN and a new memory MEM is added. The state variables and memories used in the example have the following meaning:

VRR sequence number of next expected in sequence data packet at the receiver
VRN sequence number of next expected retransmitted data packet at the receiver (new state variable)
VRH sequence number of highest expected data packet at the receiver
MEM receiver memory containing content of status message
SN sequence number of received data packet

[0047] A data packet is identified by its sequence number. Without considering the modulo operation with the maximum sequence number, the condition $VRR \leq VRN \leq VRH$ applies to the variables in the receiver. Data packets with sequence numbers $SN \geq VRH$ are first transmissions of a data packet, data packets with sequence numbers $SN < VRR$ have been successfully transmitted and acknowledged by the receiver while data packets with sequence numbers SN between VRR and VRH are retransmissions.

[0048] Figure 4 shows a procedure for a fast detection of erroneous retransmissions and to initiate a fast retransmission of said data packets. Whenever a message STATUS is initiated in step 1, the state variable VRN is set to the value of variable VRR. The content of the message STATUS is stored in memory MEM. If more than one STATUS message is transmitted during a round-trip time, multiple memories MEM are necessary, one for each open STATUS message, or the contents of several status messages are stored together in one memory. The first option is preferable, especially as it simplifies the handling if a message STATUS is lost.

[0049] The receiver enters a waiting state for data packets. When a data packet is received in step 2, the sequence number SN is determined. The sequence number SN is then compared in step 3 to the variables VRH and VRR. If the condition $(SN < VRH)$ and $(SN \geq VRR)$ applies, the data packet is a retransmission of a packet sent before and requested with the message STATUS in step 1. In this case, a round trip time has passed since the message STATUS was sent. From the measurement of this interval, optionally the round trip time is estimated in step 3a and timers which depend on the round trip time are adjusted.

[0050] In step 3b, the sequence number of the received data packet is compared to the next expected retransmission stored in variable VRN. If both values differ, the retransmissions between VRN and SN are lost.

In this case, the lost data packets with sequence numbers from VRN up to SN are marked to be requested for retransmission in corresponding message STATUS_re and the data packet with sequence number SN is marked for acknowledgement in a message STATUS_re. The variable VRN is updated and set to the sequence number of the next missing data packet in memory MEM. Optionally, the message STATUS_re for lost retransmissions can immediately be sent. Alternatively, a timer to prohibit the sending of the message STATUS_re can be more suitable. The advantages of the different options depend on the configuration of the ARQ protocol.

[0051] If the result of the comparison in step 3b is that the expected retransmission was received, the sequence number SN is marked in step 3c for acknowledgment in the next message STATUS_re. The variable VRN is set to the sequence number of the next missing data packet stored in the memory MEM. If the expected retransmission is a successful transmission of the next in sequence data packet, i.e. the data packet with the lowest sequence number still missing, the reception window of the receiver can be shifted. For this purpose, variable VRR is set to the value of variable VRN, i.e. the next missing data packet in the sequence.

[0052] If the comparison of step 3 has the result that the sequence number SN is equal or larger than variable VRH new data packets are transmitted for the first time and step 4 is initiated. In step 4, those erroneous retransmissions are detected which have sequence numbers between variable VRN and VRH. Correspondingly, retransmissions with a sequence number larger than VRN are marked in the memory MEM for further retransmission. The value of variable VRN is reset to VRR, i.e. the next expected retransmission is the first data packet in the sequence still missing. A message STATUS_re is sent as soon as possible to initiate a fast retransmission of erroneous retransmissions. The execution of step 4 can be omitted if the value of variables VRR and VRN are already equal. This condition is not necessary but advantageous as it avoids that step 4 is repeated if several data packets with sequence numbers larger than VRH are received.

[0053] The above embodiments admirably achieve the objects of the invention. However, it will be appreciated that departures can be made by those skilled in the art without departing from the scope of the invention which is limited only by the claims.

Claims

1. Method for a transmission of data packets from a transmitter to a receiver, wherein the receiver performs a first check whether a data packet is correctly received and sends a request for a retransmission of erroneous data packets to the transmitter, **characterized in that**

the receiver performs a second check whether a received data packet (DP) is a retransmission and the result of the second check is evaluated to adapt an operating parameter of the receiver or determine an operating parameter.

2. Method according to claim 1, wherein a further request (STATUS') for a retransmission is performed according to the result of the second check
3. Method according to claim 1 or 2, wherein data packets (DP) are identified by a sequence number (SN) and wherein a first variable (VRR) identifying the first data packet (DP) in the sequence requested for retransmission is stored and a data packet (DP) is rejected if the sequence number (SN) is smaller than the first variable (VRR) or larger than the first variable (VRR) plus a specified offset.
4. Method according to any preceding claim, wherein data packets (DP) are identified by a sequence number (SN) and wherein the receiver stores a further variable (VRH) identifying the last data packet (DP) in the sequence which was correctly received, wherein the second check is performed by determining whether the sequence number (SN) of a received data packet is smaller than the sequence number (SN) of said further variable (VRH).
5. Method according to any preceding claim, wherein a timer or counter determining the sending of a further request (STATUS) for a retransmission is set according to the result of the second check.
6. Method according to any preceding claim, wherein a further check is performed whether a retransmitted data packet (DP) is erroneous.
7. Method according to claim 6, wherein the further check comprises a detection whether the sequence number (SN) of a successfully received data packet (DP) is equal to the first variable (VRR).
8. Method according to claim 6 or 7, wherein at least one condition for sending a request (STATUS) for the retransmission of an erroneous retransmitted data packet (DP) differs from the conditions for sending a request (STATUS) for the retransmission of an erroneous data packet (DP) which was transmitted for the first time.
9. Method according to claim 8, wherein the sending of a request (STATUS) for retransmission is triggered by the detection of an erroneous retransmitted data packet (DP).
10. Method according to any preceding claim, wherein a request (STATUS) for a further retransmission is

suspended while a retransmitted data packet (DP) is received.

11. Method according to any preceding claim, wherein the receiver stores in a memory (MEM) which data packets are requested for retransmission. 5
12. Method according to claim 11, wherein the second check is performed by comparison of stored and received sequence numbers (SN). 10
13. Method according to claim 12, wherein the sequence number (VRN) of the next requested data packet (DP) is determined and a retransmission is defined as erroneous if the sequence number (SN) of the next received data packet (DP) differs from the sequence number (VRN) of the next requested data packet (DP). 15
14. Method according to any preceding claim, wherein the method is performed in a mobile communication system. 20
15. Receiver in a mobile telecommunication system with a reception unit for receiving data packets from a transmitter and a processor for the detection of erroneous data packets, wherein the processor initiates a request to the transmitter for a data packet (DP) detected as erroneous, characterized in that the receiver performs a method according to any preceding claim. 25 30
16. Program unit on a data carrier or loadable into a processing system of a receiver, said unit comprising code for performing the steps of a method according to any of the claims 1 to 14. 35

40

45

50

55

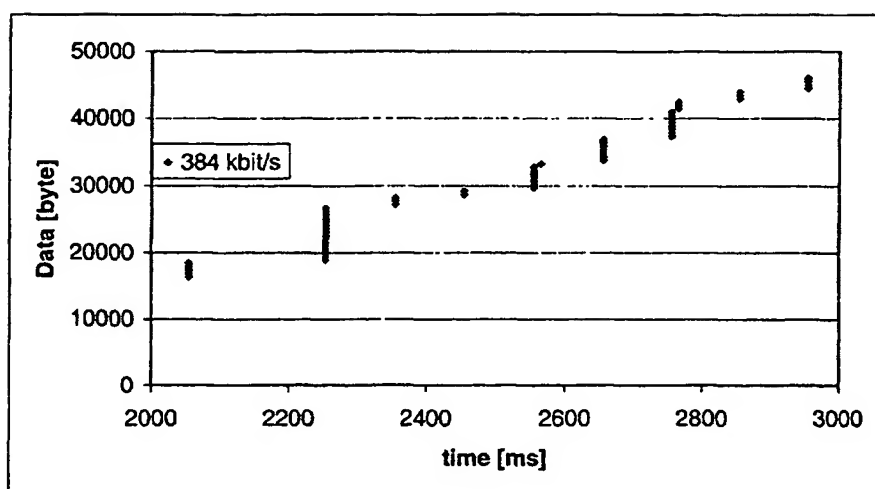


Fig. 1

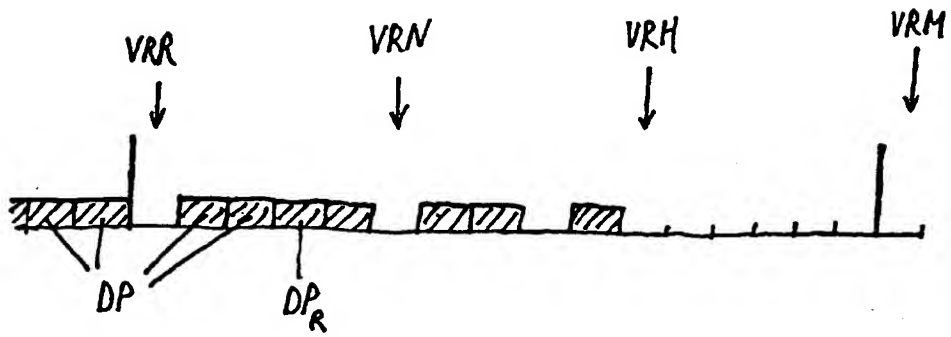


Fig 2

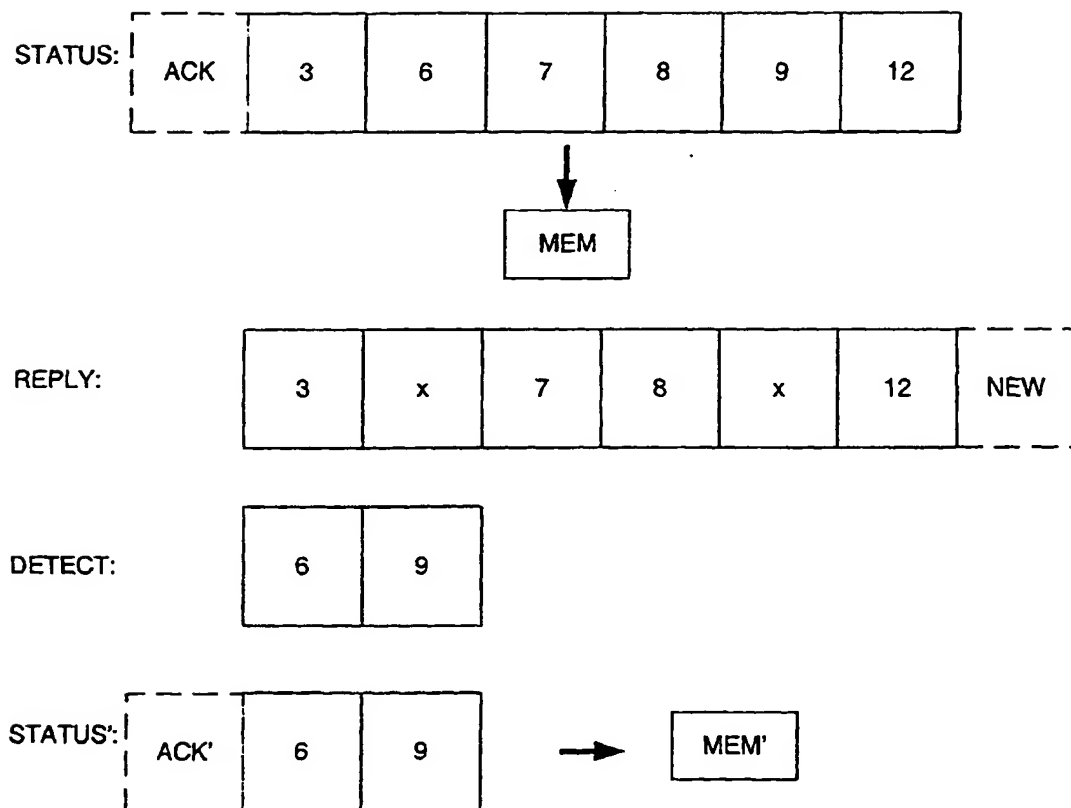


Fig. 3

```

1   Send STATUS message
    Store STATUS in memory MEM
    Set VRN=VRR

2   Receive data packet with sequence number SN

3   if (SN<VRH) AND (SN>=VRR) {
3a  Estimate the RTT and adapt timers which depend on RTT.
3b  if (SN>VRN) {
        mark lost data packets between VRN up to SN for
        retransmissions in STATUS_re
        mark SN for acknowledgement in STATUS_re
        set VRN = next missing data packet in MEM
        [Option]: send STATUS_re for lost retransmissions
    }
3c  else if (SN=VRN) {
        mark SN for acknowledgment in STATUS_re
        set VRN= next missing data packet in MEM
        if (SN=VRR) {
            set VRR=VRN }
    }
}

4   else if (SN>=VRH) and (VRN≠VRR){
    mark retransmissions >= VRN in MEM for further
    retransmission
    set VRN=VRR
    send STATUS_re to retransmit erroneous retransmissions
}

```

Fig. 4



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 00 11 1296

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 5 664 091 A (KEEN HENRY D) 2 September 1997 (1997-09-02) * page 9, column 2, line 3 - line 14 * * page 9, column 1, line 10 - line 13 * * page 9, column 2, line 43 - line 59 * * page 10, column 1, line 33 - line 36 * * page 10, column 4, line 40 - line 43 * * page 12, column 7, line 17 - line 20 * * page 13, column 9, line 17 - line 23 * * page 13, column 10, line 41 - line 55 * * page 14, column 12, line 22 - line 26 * * page 15, column 13, line 2 - line 17 *	1-13,16	H04L1/18
X	EP 0 820 167 A (MATSUSHITA ELECTRIC IND CO LTD) 21 January 1998 (1998-01-21) * page 2, column 1, line 15 - line 38 * * page 2, column 3, line 54 - line 55 * * page 4, column 1, line 32 - line 58 * * page 4, column 2, line 53 - line 58 * * page 5, column 7, line 2 - line 22 * * page 6, column 10, line 32 - line 50 * * page 7, column 11, line 42 - line 50 * * claims 1-3 *	1-15	TECHNICAL FIELDS SEARCHED (Int.Cl.7) H04L
A	EP 0 658 028 A (AT & T CORP) 14 June 1995 (1995-06-14) * page 3, column 4, line 25 - line 28 * * page 4, column 6, line 1 - line 8 * * page 7, column 12, line 44 - line 55 * * page 8, column 13, line 1 - line 36 *	1-16	
A	US 5 754 754 A (DUDLEY JOHN G ET AL) 19 May 1998 (1998-05-19) * page 10, column 3, line 29 - line 42 * * page 16, column 16, line 4 - line 7 * * claims 6,12 * --- -/--	1-13,16	
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 6 November 2000	Examiner Moreno, M
CATEGORY OF CITED DOCUMENTS X: particularly relevant if taken alone Y: particularly relevant if combined with another document of the same category A: technological background O: non-written disclosure P: intermediate document		T: theory or principle underlying the invention E: earlier patent document, but published on, or after the filing date D: document cited in the application L: document cited for other reasons --- &: member of the same patent family, corresponding document	

EPO FORM 1503 (03.02.99) (P04001)



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 00 11 1296

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	PATENT ABSTRACTS OF JAPAN vol. 018, no. 646 (E-1641), 8 December 1994 (1994-12-08) & JP 06 252978 A (TOSHIBA CORP), 9 September 1994 (1994-09-09) * abstract *	1	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 6 November 2000	Examiner Moreno, M
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons A : member of the same patent family, corresponding document			

EPO FORM 1503 (03.02) (P01C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 00 11 1296

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

06-11-2000

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5664091	A	02-09-1997	NONE		
EP 0820167	A	21-01-1998	CN	1171678 A	28-01-1998
			JP	10084336 A	31-03-1998
			US	5896402 A	20-04-1999
EP 0658028	A	14-06-1995	US	5444718 A	22-08-1995
			AU	672536 B	03-10-1996
			AU	7900894 A	08-06-1995
			CA	2132649 A,C	31-05-1995
			CN	1111889 A	15-11-1995
			JP	7202856 A	04-08-1995
			KR	149899 B	15-05-1999
US 5754754	A	19-05-1998	NONE		
JP 06252978	A	09-09-1994	NONE		

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82